DEVELOPMENT OF A SOFTWARE SAFETY PROCESS AND A CASE STUDY OF ITS USE

Annual Progress Report
Grant No. NAG-1-1123

August 1, 1996 - July 31, 1997

Submitted to:

National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23681-0001

Attention: Dr. Dave E. Eckhardt, MS 478

Submitted by:

J. C. Knight
Professor

SEAS Report No. UVA/528344/CS98/107
October 1997

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF
ENGINEERING
& APPLIED SCIENCE

University of Virginia
Thornton Hall
Charlottesville, VA 22903

INTRODUCTION

This is the annual report for the period August 1, 1996 to July 31, 1997 for NAG-1-1123. The principal investigator on this grant is Dr. John C. Knight of the Computer Science Department, University of Virginia, Charlottesville, Virginia 22903.

Research in the year covered by this reporting period has been primarily directed toward the following areas:

- Formal specification of user interfaces.
- Fault-tree analysis including software.
- Evaluation of formal specification notations.
- Evaluation of formal verification techniques.
- Expanded analysis of the shell architecture[†] concept.
- Development of techniques to address the problem of information survivability.
- Development of a sophisticated tool for the manipulation of formal specifications written in Z.

This report summarizes activities under the grant. The technical results relating to this grant and the remainder of the principal investigator's research program are contained in various reports and papers.

The remainder of this report is organized as follows. In the next section, an overview of the project is given. This is followed by a summary of accomplishments during the reporting period and details of students funded. Seminars presented describing work under this grant are listed in the following section, and the final section lists publications resulting from this grant.

[†]. The architecture previously referred to as a security kernel is now referred to as a shell architecture. The term shell better characterizes the basic form of the architecture.

OVERVIEW

The goal of this research is to continue the development of a comprehensive approach to software safety and to evaluate the approach with case studies. The case studies are a major part of the project, and they involves the analysis of specific safety-critical systems. The particular applications being used were selected because of the availability of suitable candidate systems. We consider the results to be generally applicable and in no way particularly limited by the domains.

With more and more important functions in existing and proposed safety-critical systems being implemented by computers, concern over the role of software in such systems has increased. An especially important area is that class of systems for which safety rather than reliability or availability is the overriding issue. Some research that addresses the safety of software specifically has been reported but many open questions remain. In particular, no complete process is available for engineers to follow when building applications software for systems in which safety considerations dominate. We are developing such a process through a combination of theoretical and empirical research.

The research is concentrating on issues raised by the specification and verification phases of the software lifecycle. The theoretical research is based on our framework of definitions for software safety in which the problem is broken down into *specification safety* and *implementation safety.*

In the area of specification, the main topics being investigated are:

- the formal specification of complex user interfaces,
- the evaluation of formal specification notations,
- development of tools to support the creation of specifications, and
- the development of rigorous techniques for the preparation of software safety specifications based on the analysis of system fault trees.

A second area of theoretical investigation is the development of verification methods tailored to the characteristics of safety requirements. Verification of the correct implementation of the safety specification is central to the goal of establishing safe software. In the area of specification, the main topics being investigated are:

- the evaluation of formal verification in a typical industrial context, and
- the development of system architectures that facilitate the establishment of critical system properties.

The empirical component of this research is focusing on case studies in order to provide detailed characterizations of the issues as they appear in practice, and to provide a testbed for the evaluation of various existing and new theoretical results, tools and techniques. The system that is the current focus of our work is the *University of Virginia's research nuclear reactor* (UVAR). The overall, long term approach being taken in the empirical research is to develop fully functional software of sufficient quality to be suitable for safety-critical use. This approach is necessary to ensure that the research undertaken is not weakened by unrealistic assumptions or restrictions. The empirical research is implementing the various techniques resulting from the theoretical research and using these implementations to assess the theoretical results.

The focus during the reporting period has been the UVAR case study. Despite the body of existing work, the exploitation of computers in nuclear control systems is not extensive. More specifically, the use of formal software specifications has been undertaken only rarely even though it is well known that specification errors are the most common types of error in safety-critical systems and that formal specifications are capable of far greater precision than natural language. A good example of careful specification is the work of Parnas on the Darlington project. An example of the use of informal specification in a modern system is Sizewell B in the United Kingdom in which the entire system is specified in natural language.

The goal is to determine the utility of formal specifications in digital nuclear system. With the help of members of the staff of the University's reactor facility and the Department of Mechanical, Nuclear and Aerospace Engineering, experiments are being undertaken in which formal specifications are being prepared for parts of an advanced reactor control system.

The control requirements of the reactor are being studied to determine the requirements for emergency shutdown, monitoring and operation, and safe operation including the response to a variety of equipment failures. System fault trees have been developed and are being refined for certain parts of the system to permit detailed documentation of the software's failure response requirements to be acquired.

Formal specifications for the different control requirements have been prepared in Z, PVS, Statecharts, and SCR/A7. As part of the assessment of the techniques for nuclear applications, experiments have been undertaken to gauge the utility of the notations to nuclear engineers and others. These assessment experiments are being continued and extended.

Research results to date are documented in various papers and reports, and they are not repeated here. Copies of these papers and reports have been supplied to the sponsor under separate cover.

ACCOMPLISHMENTS DURING REPORTING PERIOD

The accomplishments in the various activity areas are summarized in this section. More details of the work in the different activity areas are contained in the publications listed in a later section of this report.

Although much of the work undertaken during the reporting period has focused on the nuclear-reactor case study, this is not a limitation in any sense because our goal is research in software engineering for control systems in general. The proposed control system for the reactor is quite typical in its software requirements.

User Interface Specification

Research has continued on the formal specification of user interfaces. This work is being pursued in cooperation with Dr. S. Brilliant of Virginia Commonwealth University (Richmond, VA). The user interface for the University of Virginia Reactor (UVAR) has been completed, and the approach to user-interface specification has been evaluated. The possibility of automated analysis of the user-interface specification is being investigated.

Recent work in this area has been documented in a conference paper (see list of publications).

Comprehensive Fault-Tree Analysis

The system fault tree is the primary model that is used to determine the risks associated with the various system hazards. Fault-tree analysis is used to refine system designs to permit reductions in risk levels where these levels are above acceptable thresholds. For systems that involve software, the introduction of software events into fault trees has proved problematic. The issue is the difficulty of quantifying failure probabilities for software.

Since the fault-tree model is the primary technique for risk analysis, it is essential that it be possible to analyze fault trees for systems that depend on software. In addition, it should be noted that the fault tree can and should have a heavy influence on the specification for the software for a system. Thus a system fault tree is a primary input to software safety specification.

We have developed a number of techniques for the inclusion of software items into fault-tree analysis. Recent work in this area has been documented in a conference paper (see list of publications).

Formal Specification

We have completed an assessment of formal specification based on a comprehensive framework for evaluation developed previously. This evaluation was designed to permit the necessary research directions to be determined that will allow more extensive application of formal specification in industry. Results in this area have been documented in a conference paper and a technical report (see list of publications)

Formal Verification

As with formal specification, we have completed an assessment of formal verification based on a comprehensive framework for evaluation developed previously. This evaluation was designed to permit the necessary research directions to be determined that will allow more extensive application of formal verification in industry. Results in this area have been documented in a technical report (see list of publications).

Shell Architecture

We have begun to study the expanded role of shell architectures in modern distributed systems. The goal is to adapt and expand the results obtained with safety kernels to systems implemented on distributed targets, and to systems that employ commercial off-the-shelf (COTS) components. Preliminary results indicate the utility of shells in this expanded area but that new types of shell are required.

Information Survivability

Many important safety-critical systems are in fact information systems. The simplest example is air-traffic-control systems but many similar systems exist in other domains. In addition, modern embedded control systems (such as avionics) are being integrated into more complex "systems of systems" making survivability an issue even for embedded control systems.

We have begun research into the general problem of information system survivability. The topics being investigated encompass many of the topics already of interest under this grant. In addition, security issues are being investigated. This work is being performed in cooperation with Dr. John McHugh of Portland State University.

Software Tools

In cooperation with Odyssey Research Associates (ORA) Canada, Ltd. we have begun the development of a sophisticated tool for the manipulation of formal specifications written in Z. ORA has developed a system, Z/EVES, that performs syntax analysis and type checking of Z together with a powerful theorem prover. This system is used with an ASCII representation of the Z character set that is somewhat inconvenient.

The tools we are developing are based on Microsoft Office packages that provide powerful facilities through a high quality user interface. Using this system, the user is no longer required to manipulate Z specifications through the ASCII interface.

SUPPORTED STUDENTS

During the reporting period, the following students were supported in whole or in part under this grant:

Name	-	Luis G. Nakano
Dissertation Title	-	Techniques for the Design and Safety-Analysis of Computer-Based Systems.
Degree	-	Ph.D.
Status	-	In progress
Name	-	Colleen Dejong
Thesis Title	-	Formal Specification: A Systematic Evaluation
Degree	-	M.S.
Status	-	Graduated May, 1997
Name	-	Matthew Gibble
MCS Project Title	-	Formal Verification: An Evaluation
Degree	-	M.C.S.
Status	-	Graduated May, 1997
Name	-	Roy Bodalya
BS Thesis Title	-	Using Software Reuse to Develop a Z Specification Tool
Degree	-	B.S.
Status	-	Graduated May, 1997
Name	-	Meng Yin
BS Thesis Title	-	The Front End of Software Development: Implementation of a Formally Specified User Interface
Degree	-	B.S.
Status	-	Graduated May, 1997
Name	-	Steve Ziegler
BS Thesis Title	-	Determining the Feasibility of Building a Windows Z Specification Tool
Degree	-	B.S.
Status	-	Graduated May, 1997
Name	-	Zachariah Kohn
BS Thesis Title	-	TBD
Degree	-	B.S.
Status	-	In progress

PRESENTATIONS GIVEN

Seminars (not including conference presentations) describing the research being performed under this grant were presented at the following institution during the reporting period:

- Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA.

PUBLICATIONS

During the reporting period, the following papers and documents were prepared from the principal investigator's research program[†]:

1. Knight, John C. and Michael F. Dunn, "Software Quality through Software Reuse", *Annals of Software Engineering*, to appear.
2. Ammann, Paul, Dahlard L. Lukes, and John C. Knight, "Applying Data Redundancy to Differential Equation Solvers", *Annals of Software Engineering*, to appear.
3. Knight, John C., Colleen L. DeJong, Matthew S. Gibble, and Luis G. Nakano, "Why Are Formal Methods Not Used More Widely?", *Proceedings: Fourth NASA Formal Methods Workshop*, Hampton, VA (September 1997)
4. Knight, John C. and Luis G. Nakano, "Software Test Techniques for System Fault-Tree Analysis", *Proceedings: SAFECOMP '97*, York, UK (September 1997).
5. Knight, John C., "Is Information Security An Oxymoron?", *Proceedings: COMPASS '97* (Panel Presentation), Gaithersburg, MD (June 1997).
6. Ying, Meng., "The Front End of Software Development: Implementation of a Formally Specified User Interface", B.S. Thesis, School of Engineering and Applied Science, University of Virginia, Charlottesville, VA 22903 (May 1997)
7. DeJong, Colleen L., Matthew S. Gibble, John C. Knight, and Luis G. Nakano, "Formal Specification: A Systematic Evaluation", Technical Report CS-97-09, Department of Computer Science, University of Virginia, Charlottesville, VA 22903 (May 1997).
8. DeJong, Colleen L., Matthew S. Gibble, John C. Knight, and Luis G. Nakano, "Formal Verification: An Evaluation", Technical Report CS-97-13, Department of Computer Science, University of Virginia, Charlottesville, VA 22903 (May 1997).
9. Knight, John C. and Susan S. Brilliant, "Preliminary Evaluation of a Formal Approach to User Interface Specification", *Proceedings: Tenth International Conference of Z Users*, LNCS 1212 Springer Verlag, Reading, UK (April 1997)
10. Knight, John C. and Kevin J. Sullivan, "Survivability Architectures", *Proceedings: ISW '97—Information Survivability Workshop*, San Diego (March 1997).
11. Sullivan, Kevin J., John C. Knight, Jake Cockrell, and Shengtong Zhang, "Product Development With Massive Components, *Proceedings: Twenty-First Annual Software Engineering Workshop*, Greenbelt MD (December 1996)

[†]. This list includes all publications attributed to all sponsors.

DISTRIBUTION LIST

1 - 3	Dr. Dave E. Eckhardt, MS 478 National Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-0001 (804) 864-1698
4	Mr. Joseph S. Murray, Grants Officer, MS 126 Acquisition Division National Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-0001 (804) 864-7709
5 - 6*	National Aeronautics and Space Administration Scientific and Technical Information Facility P. O. Box 8757 Baltimore/Washington International Airport Baltimore, MD 21240
7 - 8	J. C. Knight
9	J. Stankovic
10 - 11	M. Rodeffer
**	SEAS Postaward Research Administration
12	SEAS Preaward Research Administration

*1 unbound copy
**Cover letter

JO#7845:ph